

Vysoká škola báňská – Technická univerzita Ostrava

Fakulta elektrotechniky a informatiky

Katedra telekomunikační techniky

Infrastrukturní ENUM v rozsáhlých sítích

Infrastructural ENUM in Large Networks

2015

Rostislav Cáb

Zadání bakalářské práce

Student: **Rostislav Cáb**

Studijní program: B2647 Informační a komunikační technologie

Studijní obor: 2601R013 Telekomunikační technika

Téma: **Infrastrukturní ENUM v rozsáhlých sítích**
Infrastructural ENUM in Large Networks

Zásady pro vypracování:

Bakalářská práce se v teoretické části zabývá obecnému popisu propojování telefonních sítí VoIP operátorů a to jednak centralizovanému tak i decentralizovanému řešení. Detailně popisuje decentralizované řešení pomocí ENUM stromu a dokumentuje praktickou implementaci této technologie mezi pobočkovými ústřednami Asterisk.

Cíle práce:

1. Teoretický rozbor tematiky propojování VoIP operátorů
2. Popis propojení pomocí technologie ENUM
3. Praktická konfigurace propojení ústředn Asterisk pomocí infrastrukturního ENUM stromu
4. Automatizace řešení pomocí instalačních skriptů
5. Ověření funkčnosti propojení pomocí rozboru signalizace

Seznam doporučené odborné literatury:

R. Bryant, L. Madsen, J. Meggelen, Asterisk: The Definitive Guide, O'Reilly Media; Fourth Edition, 2013.

Uzelac, A., Ed., and Y. Lee, Ed., "Voice over IP (VoIP) SIP Peering Use Cases", RFC 6405, November 2011.

Faltstrom, P. and M. Mealling, "The E.164 to Uniform Resource Identifiers (URI) Dynamic Delegation Discovery System (DDDS) Application (ENUM)", RFC 3761, April 2004.

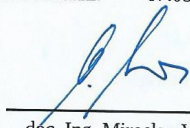
M. Vozňák, Voice over IP. Vysokoškolská skripta, 176 stran. Vydavatel: VŠB-TU Ostrava, Dotisk prvního vydání, v Ostravě, září 2009, ISBN 978-80-248-1828-3

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

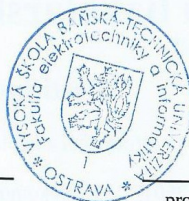
Vedoucí bakalářské práce: **Ing. Martin Mikulec**

Datum zadání: 01.09.2014

Datum odevzdání: 07.05.2015



doc. Ing. Miroslav Vozňák, Ph.D.
vedoucí katedry




prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlášení studenta

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě dne: 7.5.2015



podpis studenta

Poděkování

Rád bych poděkoval Ing. Martinu Mikulcovi za odbornou pomoc a konzultaci při vytváření této bakalářské práce. Dále bych rád poděkovat své rodině a přítelkyni za podporu.

Abstrakt

Tato bakalářská práce se zabývá implemetací systému ENUM do prostředí Asterisku. ENUM neboli „E164 Number Mapping“ je systém mapování telefonních čísel do prostředí Internetu. Centrální národní databáze ENUM telefonních čísel svázaných s VoIP identifikátory je velmi chudá. Nicméně využití systému ENUM v privátních sférách je více zajímavější. Cílem této bakalářské práce je navrhnout funkční řešení spojení dvou pobočkových ústředen Asterisk. Poté toto funkční řešení zautomatizovat pomocí instalačního skriptu a v posledním kroku provést rozbor funkčnosti propojení pomocí rozboru signalizace.

Klíčová slova

VoIP; SIP; ENUM; DNS

Abstract

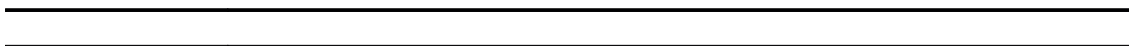
This bachelor thesis deals with the implementation of ENUM to the Asterisk environment. ENUM or "E164 Number Mapping" is a system of mapping telephone numbers into the Internet environment. A central national database of ENUM phone numbers tied to VoIP identifier is very poor. However, the use of ENUM in the private sphere is more interesting. The aim of this work is to propose workable solutions connecting two Asterisk PBX. Then, this functional solution using automate the installation script and in the last step to analyze the functional connections using signaling analysis.

Key words

VoIP; SIP; ENUM; DNS

Seznam použitých zkratek

Zkratka	Význam
VoIP	Voice over Internet Protocol
TCP/IP	Transmission Control Protocol / Internet protocol
UDP	User Datagram Protocol
PCM	Pulse code modulation
SIP	Session Initiation Protocol
IAX	Inter Asterisk eXchange
ITU-T	International Telecommunication Union – Telecommunication Standardization Sector
PSTN	Public Switched Telephone Network
ENUM	E.164 Number Mapping
DNS	Domain Name System
TLD	Top Level Domain
URI	Uniform Resource Identifier
FQDN	Fully Qualified Domain Name
SRV	Service record
NAPTR	Name Authority Pointer
PBX	Private Branch Exchange
BASH	Bourne Again Shell



Obsah

Úvod.....	-11-
1. Teoretický rozbor tematiky propojování VoIP operátorů.....	-12-
1.1 Aktuální situace v České Republice.....	-12-
1.2 VoIP.....	-13-
1.2.1 Princip VoIP.....	-13-
1.3 Signalizační protokoly VoIP.....	-14-
1.3.1 SIP.....	-14-
1.3.2 H.323.....	-15-
1.3.3 IAX.....	-16-
2. Popis propojení pomocí technologie ENUM.....	-17-
2.1 Přehled základních pojmů.....	-17-
2.1.1 E.164 Number Mapping.....	-17-
2.1.2 Domain Name System.....	-17-
2.2 Umístění telefonních čísel do doménového stromu.....	-17-
2.3 ENUM v ČR.....	-18-
2.4 Infrastrukturní ENUM.....	-19-
2.5 Mapování mezi adresovými prostory.....	-19-
2.6 Jak ENUM funguje ?.....	-20-
2.6.1 Převod telefonního čísla na plně kvantifikované doménové jméno.....	-21-
3. Praktická konfigurace propojení ústředěn Asterisk pomocí ENUM.....	-22-
3.1 Asterisk.....	-22-
3.1.1 sip.conf.....	-22-
3.1.2 extension.conf.....	-23-
3.2 Bind.....	-25-
3.2.1 named.conf.....	-25-
3.2.2 db.666.....	-25-
3.3 Registrace na softphonech.....	-25-
3.3.1 Yate.....	-26-
3.4 Shrnutí.....	-26-
4. Automatizace řešení pomocí instalačního skriptu.....	-28-
5. Ověření funkčnosti propojení pomocí rozboru signalizace	-29-
5.1 Wireshark.....	-29-

5.2 Rozbor ENUM.....	-29-
Závěr.....	-32-
Použitá literatura.....	-33-
Seznam příloh.....	-34-

Úvod

Tato bakalářská práce pojednává o technologii ENUM a její implementaci do ústředny Asterisk. Technologie ENUM by se dala stručně popsat jako mapování telefonních čísel do prostředí Internetu. Spojuje tedy dva dosud oddělené světy, svět tradičního telefonování prostřednictvím telefonních sítí, na které je většina uživatelů zvyklá a svět Internetu se všemi svými výhodami. Pro velké množství uživatelů by technologie ENUM mohla přinést mnoho výhod, pokud by je uživatel dokázal identifikovat a následně využít pro svůj prospěch. Tato bakalářská práce může být vstupním bodem pro pochopení celé problematiky technologie ENUM. Pro eventuální zájemce může tato bakalářská práce sloužit jako vodítko k praktické realizaci.

První kapitola popisuje aktuální oblasti telefonní komunikace. Jedna z oblastí je telefonování přes Internet a tímto směrem se dál bude tato práce odvíjet. Hlavní zástupce technologií volání přes Internet je technologie VoIP. Vysvětlen je hlavně popis a dále funkčnost této technologie. Technologie VoIP využívá pro svou funkčnost signální protokoly. Tyto protokoly SIP, H.323 a IAX jsou v kapitole popsány z hlediska jejich funkčnosti.

V druhé teoretické kapitole je vysvětleno, jaký je princip funkčnosti technologie ENUM. Technologie ENUM má velmi blízko technologii DNS, proto následující část kapitol tyto dvě technologie porovnává a přibližuje pochopení celé problematiky.

Třetí kapitola všechny tyto teoretické poznatky využívá a na dvou serverech si nainstaluje programy Asterisk a Bind. Pomocí těchto programů se nastaví dvě pobočkové ústředny, které budou využívat technologii ENUM, přes kterou si také budou moct uzavřít spojení.

Čtvrtá kapitola předchozí krok zautomatizuje pomocí instalačního skriptu.

Pátá kapitola otestuje funkční propojení pomocí rozboru signalizace v programu Wireshark. Kapitola popisuje funkční spojení dvou pobočkových ústředí přes ENUM technologii.

1. Teoretický rozbor tematiky propojování VoIP operátorů

1.1 Aktuální stav v České Republice

V České Republice je nejvíce rozšířený trend mobilních operátorů. Uživatel obdrží od svého operátora jednoznačné telefonní číslo, které má ve správě poskytovatel telefonních služeb a ten má na starost všechny příchozí a odchozí hovory na toto telefonní číslo. Všichni poskytovatelé jsou navzájem propojeni a účtují si poplatky za propojení. Tyto poplatky se samozřejmě projevují na koncové ceně zákazníkovi.

Pro přenos hlasu přes IP protokol je důležitá technologie přes, kterou se bude IP protokol přenášet. V dnešní době máme více technologií, které můžeme využít např. Email, instant messaging, VoIP nebo videokonference. Hlavní myšlenkou je, aby koncový uživatel využívající těchto služeb byl dostupný pod jedním identifikátorem, který bude zastupovat tyto služby. Nejvíce používaným identifikátorem je uživatel@doména.tld, který dobře známe z emailové komunikace. Výhodou je, že se celé portfolio služeb se právě může schovávat pod jedním identifikátorem. Uživatelům v dané doméně nabízí služby poskytovatel a je jen na něm, jaké služby jim nabídne. Služby jako poslat email, kontaktovat uživatele prostřednictvím chatu, zavolat mu, poslat mu SMS nebo fax je možné pomocí stejného identifikátoru uživatel@doména.tld.

Z praxe můžeme zaznamenat dva případy toho, jak se v dnešní době využívá služby VoIP.

Prvním případem jsou uživatelé internetových služeb, které nabízejí volání zdarma v rámci své sítě, např. přes Facebook, Skype, Viber nebo Google Hangout. Tato možnost je nenáročná a pohodlná. Stačí si ve svém počítači nainstalovat jednu z těchto aplikací, zaregistrovat se a propojit se přes Internet s dalšími uživateli a navzájem si volat, posílat sms a další možnosti. Tyto služby mohou využívat i korporátní sféry, kdy se tímto způsobem připojují ke své firmě a tím zvyšují bezpečnost komunikace a šetří náklady firmy. Zde je tedy vidět zvyšující se podíl VoIP technologií a její prolínání se s uzavřeným světem operátorů.

Druhým případem je, že VoIP poskytovatel nám nabízí služby volání. Velkou výhodou je, že poskytovatel tyto hovory může směřovat bezplatně a bezpečně. Takové spojení nazýváme peering, které se zařazuje do svého směrovacího plánu SIP proxy. Tímto se vytvoří svoje vlastní síť, tudíž se hovory nebudou směřovat prostřednictvím veřejné sítě, ale pomocí vytvořeného peeringu přes IP síť. Takto poskytovatel může uživatelům nabízet výhodné podmínky. V České Republice se tato možnost využívá a na trhu je 34 VoIP operátorů, kteří využívají SIP peeringu.

1.2 VoIP

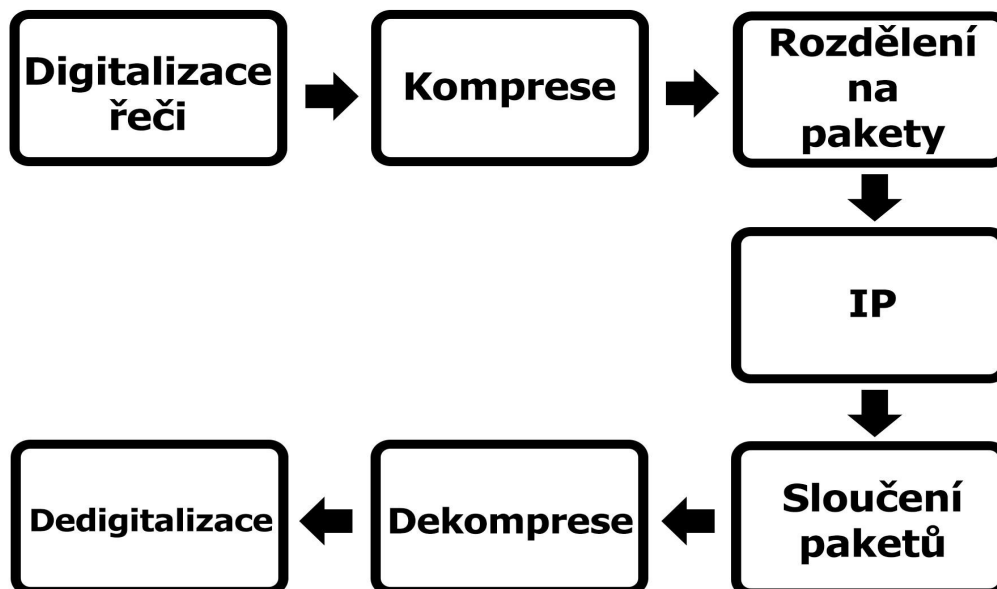
Pod názvem VoIP neboli Voice over Internet Protokol je možné si představit přenos hlasu přes IP protokol. VoIP komunikuje přes datové sítě s UDP nebo TCP/IP protokolem. VoIP byl vyvinut v roce 1995 [1], kdy byl první hovor směrován přes IP síť, firmou Vocaltec. Již v roce 1996 byla vytvořena brána, která umožnila vysílat hovor v reálném čase. VoIP byla prosazována pro svou levnější cenu v meziměstských a mezinárodních telefonních spojení, proto se stala VoIP ihned lukrativní záležitostí pro mnoho firem. VoIP pracuje na základě standardů, které byly vypracovány ITU-T. Velké množství komunikačních a internetových protokolů je využíváno pro přenos zvuku a obrazu přes internet v technologii VoIP.

Hlavní výhoda technologie VoIP je implementovatelnost všude, kde se vyskytuje internetová přípojka. Poté nám stačí připojit k multimediálnímu PC sluchátka, mikrofon a potřebné softwarové vybavení, abychom mohli využít technologii VoIP.

Kvalita hovoru v technologii VoIP je závislá na zpoždění a ztrátě dat.

1.2.1 Princip VoIP

Na obrázku 1.1 je vyobrazen systém VoIP v praxi, kdy nám jednotlivé bloky představují, jak se datové toky transformují. Šipky označují cestu od jednoho uživatele k druhému. Slova v lidské řeči tvoří akustickou vlnu, která se přetváří na binární signál a posílají je po síti. Tento krok nám zajistí digitalizace řeči. Kompresí se rozumí zmenšení objemu dat nadále rozdělovány na IP pakety, které se budou posílat prostřednictvím sítě společně s dalšími daty z jednoho PC na druhý. Na druhé straně se provede stejný postup jako při odesílání dat. Všechny rozdělené IP pakety se znovu sloučí dohromady. Provede se dekomprese a dedigitalizace dat. Díky tomuto procesu znovu získáme původní data.



Obrázek 1.1 cesta hlasu v technologii VoIP

1.3 Signalizační protokoly VoIP

Signalizační protokoly v technologii VoIP slouží pro navázání spojení. Jejich funkci je řídit tok a ukončovat spojení. Nejvýznamnější protokoly do této skupiny patří SIP, H.323 a IAX.

1.3.1 SIP

Protokol SIP je signalizační protokol, který má na starost oboustranné spojení, dohled a rozpad mezi dvěma a více účastníky komunikace. Uživatelé jsou spojováni přes SIP protokol pomocí SIP proxy serveru. Protokol SIP používá UDP port 5060, ale může taky pracovat i na TCP portu 5060, což zajišťuje bezproblémovou rozšiřitelnost a poměrně jednoduché ladění. Protokol SIP byl vyvinut v roce 1996 IETF (Internet Engineering Task Force). O tři roky poté byl specifikován v prvním RFC 2543. V současné době je na trhu verze RFC 3261.[8]

SIP protokol si našel uplatnění hlavně v internetové telefonii a vytváření multimediálních konferencí. SIP lze také využívat k spojení s PSTN a internetovou sítí. Také je schopen rozeznat, kolik je účastníků připojeno k danému hovoru a evidovat počet účastníků. SIP využívá při identifikaci v síti podporu mapování (name mapping) a služeb přesměrování (redirection service), díky tomu je hlavní vlastností protokolu SIP mobilita. Každému účastníkovi je přidělen SIP URI (SIP Uniform Resource Identifier), který je jedinečný a díky němu může využívat služby SIP kdekoli na světě. Uživatel si může vytvořit adresu SIP URI v tomto tvaru sip:user@host.

Architektura SIP

Mezi prvky SIP architektury patří

- UA (User Agent)
- Servery

User Agent

User Agents (UA) jsou koncovými zařízeními SIP sítě, kde se starají o navazování spojení s ostatními UA. SIP UA může plnit roli buď klienta tzv. UAC, který požadavky vysílá nebo může plnit roli serveru tzv. UAS, který požadavky přijímá a vrací odpověď SIP. Tyto role mají pouze po dobu SIP transakce. Na koncovém zařízení musí být implementován jak UAC, tak i UAS.[2]

Servery

Servery jsou v SIP architektuře zařízení, které zprostředkovávají spojení mezi volajícím a volaným (mezi UA). V SIP architektuře rozlišujeme tyto tři typy serveru:

1.Proxy Server

Hlavním úkolem Proxy Serveru je zajistit, aby požadavek byl poslán do jiné jednotky blíže k cílenému uživateli. Mají na starost směrování žádosti o spojení dle aktuálního umístění adresáta. Proxy server můžeme rozdělit na stateless a stateful. Stateless, nebo-li bezstavový, je poměrně jednoduchý a přeposílají pouze zprávy nezávisle na jejich vzájemné vazbě. Stateful server je mnohem komplikovanější, protože po přijetí požadavku server vytvoří záznam stavu a drží důležité informace, dokud nedojde k ukončení transakce. Využití mají pro jednoduché překládání zpráv a směrování.

2.Registrar Server

Je server, který přijímá požadavky na registraci od uživatelů a mapuje jejich aktuální polohu a ukládá informaci do lokalizační databáze(location database)

3.Redirect Server

Tento server nám umožňuje se připojit se do externí domény tzv. Přesměrováním. Server přijímá požadavky a vyhledává příjemce v lokalizační databázi vytvořené námi už dobře známým Registrat Servrem. Poté se vytvoří záznam a posílá jej odesílateli požadavku v odpovědi zařazené do třídy 3xx.

1.3.2 H.323

Doporučení H.323v1 z roku 1996 ve verzi 1 popisuje terminály, zařízení a služby pro multimediální komunikaci v lokálních počítačových sítích LAN bez garance kvality služeb. Terminály H.323 mohou přenášet hlas, data a video, nebo kombinaci služeb včetně videotelefonie. Evoluce standardu H.323 přinesla další verze, roce 1998 byla přijata verze 2,

1. Teoretický rozbor tematiky propojování VoIP operátorů

kteřá řešila především vylepšení H.245, nástroj QoS pomocí rezervačního protokolu RSVP a autentizaci standardem H.235. Další verze 3 je roku 1999 a přispěla především v oblasti dopňkových služeb, která řeší doporučení H.450. Nové služby přenáší i verze 4 z roku 2000, v roce 2003 byla uvolněna verze 5, která řeší i podporu ENUM (univerzální identifikátory, vazba DNS a E.164), z roku 2006 je verze 6 s vylepšeným zálohováním (hot standby) a poslední verze 7 je z roku 2009 [2].

V dnešní době je protokol H.323 široce implementován a to hlavně v hlasové a videokonferenční zařízení. Využívá se i v různých internetových aplikacích, kde je potřeba přenos zvuku a obrazu v reálném čase přes IP síť např. GnuGK, Ekiga a NetMeeting. Technologie QoS se stala v protokolu H.323 nedílnou součástí, která vylepšila hlasové a multimediální přenosy, a proto se protokol H.323 stal standardem pro videokonference.

1.3.3 IAX

IAX protokol byl vyvinut firmou Digium a je nejmladším protokolem používaným pro VoIP. IAX je transportní protokol, který používá pouze jeden port a to UDP (4569). Musíme si uvědomit že IAX je protokol otevřený, tudíž každý si ho může stáhnout a vyvíjet. V dnešní době existuje i druhá verze IAX2, ale jedná se pořád o stejnou nejnovější verzi IAX. Oproti SIP a H.323 protokolu má výhodu tu, že je průchodný přes NAT, tudíž odstraňuje problém s nastavením NAT. Současná verze protokolu IAX2 se od ostatních VoIP protokolů liší především tím, že signační a media data jsou přenášena jedním datovým tokem na jednotném portu v binárním formátu [2].

2.Popis propojení pomocí technologie ENUM

2.1 Přehled základních pojmů

Tato část práce popisuje základní pojmy pro lepší orientaci a pochopení celé problematiky ENUMu.

2.1.1 E.164 Number Mapping

E.164[2] je standard Mezinárodní telekomunikační unie, který definuje mezinárodní číslovací plán pro celosvětovou veřejnou síť. Tento číslicový plán nám definuje obecný formát pro mezinárodní čísla tak, že každý koncový účastník měl svůj jednoznačný identifikátor, telefonní číslo a z tohoto telefonního čísla byl dostupný z jakékoliv jiné veřejné telekomunikační sítě. Tento standard definuje formáty telefonních čísel a jejich prefixy, kde každá země má jiný prefix před devíti místným telefonním číslem. Číslicový plán je omezen na 15 číslic.

E.164 Number Mapping je internetová databáze, která mapuje telefonní identifikátory neboli telefonní čísla a jmenné identifikátory v prostředí Internetu. Tato funkce se nazývá ENUM (též eNUM, Enum, eNum). Touto funkcí vznikla spojitost mezi velmi používanými telefonními a internetovými nástroji a aplikacemi.[3]

2.1.2 Domain Name System

Je hierarchický systém doménových jmen, který překládá doménové názvy na IP adresy. DNS byl vyvinut proto, aby si člověk nemusel pamatovat jednotlivé IP adresy, což je pro člověka velmi nepraktické a náročné. Člověk si tedy musí zapamatovat pouze doménové názvy a systém DNS se postará o jejich překlad na správnou IP adresu. Proto když si chce člověk otevřít v prohlížeči Seznam.cz nemusí vypisovat celou IP adresu serveru, kde jsou tyto stránky umístěny, ale napíše jednoduše www.seznam.cz a systém DNS zajistí překlad doménových jmen na IP adresu a opačně.

Zjednodušený princip DNS systému je takový, že po vyslání požadavku na server www.seznam.cz se DNS zeptá na IP adresu tohoto doménového jména. Server DNS vyhledá v doménovém stromu a pošle patřičnou IP adresu, kterou po jejím zjištění již kontaktuje přímo server Seznam na jeho IP adrese 194.228.32.3, kterou mu DNS poskytl. Poté si prohlížeč vyžádá požadovaný obsah, který mu už server Seznam předá.

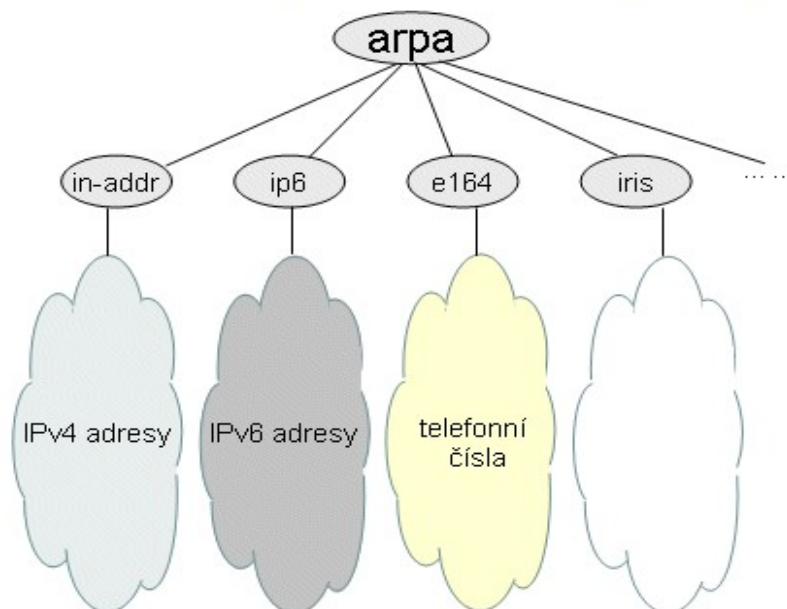
Systém DNS nemusí primárně překládat jenom doménové jména na IP adresy, ale pokud mu zadáme druh dotazu, který bude odpovídat aplikacím jako jsou např. TELNET, FTP, SMTP a SIP adresy bude DNS schopný tyto aplikace přeložit na IP adresu.

2.2 Umístění telefonních čísel do doménového stromu

V předchozích příkladech se připojila k obrácenému zápisu IP adresy nebo telefonního čísla přípona ve tvaru e164.arpa. Tento zápis znamená to, že doména nejvyšší úrovně neboli

2. Popis propojení pomocí technologie ENUM

TLD (Top Level Domain) je doména arpa a pod ní jsou subdomény SLD (Second Level Domain) neboli domény druhé úrovně jako je např. in-addr nebo e164. Pod doménu nejvyšší úrovně, tedy arpa, spadají další telefonní či jmenné prostory adres. V dnešní době pod doménu arpa nespádá jenom prostor číselných adres IP ve verzi 4 ale také prostor adres ve verzi 6. IP verze 4 spadá pod zmiňovanou subdoménu in-addr a IP verze 6 spadá pod subdoménu ip6. Pro telefonní čísla byla definována subdoména pod názvem e164. V budoucnosti se určitě můžeme dočkat nové subdomény, která bude spadat pod doménu nejvyšší úrovně.



Obrázek 2.2 Umístění subdomén v doménové hierarchii [5]

2.3 ENUM v ČR

Historie ENUM se v Čechách začala psát v červnu roku 2003, kdy Mezinárodní telekomunikační unie (ITU) přidělila sdružení CZ.NIC příslušnou národní doménu (.0.2.4.e164.arpa) pro telefonní čísla v pevných i mobilních sítích v České republice. Okamžitě na to začaly přípravy na spuštění systému ENUM. V dubnu roku 2006 pověřilo Ministerstvo informatiky České Republiky sdružení CZ.NIC provozem technologie ENUM a spuštěním testovacího provozu. Ten odstartoval 20. září 2006. Od tohoto dne si mohli zájemci registrovat své domény u dvou registrátorů (KRAXNET, IGNUM). Hovory přes ENUM podporovali ve své síti telekomunikační operátoři IPEX a CESNET (sdružení akademických institucí). Technologie ENUM byla v Česku od spuštění testovací fáze plně funkční, zkušební období sloužilo především k doladění způsobu registrace a validace domén ENUM k telefonním číslům a k ověření funkčnosti registračního systému. Testování registračního systému mělo svůj význam. Byl zde totiž poprvé využit systém FRED, ve kterém jsou spravovány všechny domény s koncovkou .cz. [6]

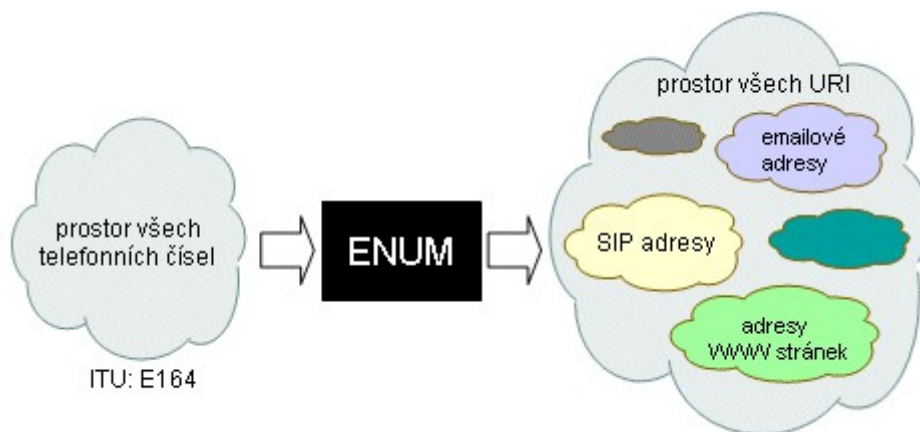
2.4 Infrastrukturní ENUM

Pojem infrastrukturní může být zaměnitelný s pojmem „operátorský“. Systém ENUM, jak už bylo zmíněno využívá systém DNS a u každé takové databáze je rozhodující, kdo má do tohoto systému přístup a jaké práva mu budou přidělena. DNS databáze je v prostředí internetu veřejně dostupná a každý si z ní může získat informace, avšak nemůže v ní dělat změny. V systému ENUM je to trochu rozdílné a jde o oprávněnost použití informace ENUM záznamu, kdo je oprávněn měnit ENUM záznamy a kdo je zadává do databáze. V tomto I-ENUM se jedná o uzavřenou formu, kde je poskytovatel oprávněn vznášet dotazy do databáze I-ENUM, který je umístěn v DNS. Tento typ I-ENUMu můžeme použít v menších firmách, kde může nahradit volání přes pevné telefony. Tento typ I-ENUMu je využit i v této práci. Na druhé straně je veřejný ENUM kde se jedná o otevřenou formu ENUMu a kdokoliv s přístupem k internetu může vznášet požadavky do databáze ENUMu, respektive do její DNS databáze.

2.5 Mapování mezi adresovými prostory

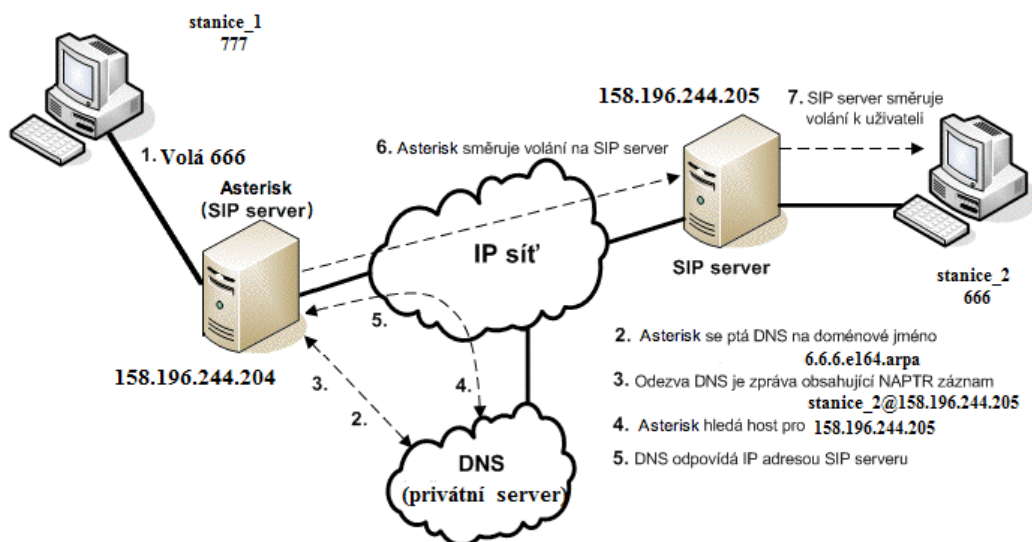
ENUM je teoreticky tabulka, která nabízí převody typu 1:1 vždy jedno telefonní číslo na jednu symbolickou adresu. Výhodou ENUMu je, že symbolická adresa nemusí být stejného typu. Symbolické adresy můžeme rozdělit do několika typů:

- SIP jako E2U+sip, používající URI např. sip:stanice_1@158.196.244.204
- H.323 jako E2U+h.323, používající URI např. h323:stanice_1@158.196.244.204
- Internet fax jako E2U+ifax, používající URI např. <mailto:fax@fax.vsb.cz>
- Telephone jako E2U+tel, používající URI např. tel:+955699169:svc=voice
- Fax jako E2U+fax:tel, používající URI např. tel:+955699169:svc=fax
- Email používající URI např. mailto: cab.rostislav.st@vsb.cz
- Web používající E2U+web:http, používající URI např. <http://www.vsb.cz>[2]



Obrázek 2.5 Mapování prostoru telefonních čísel do množin prostorů v Internetu[5]

2.6 Jak ENUM funguje ?



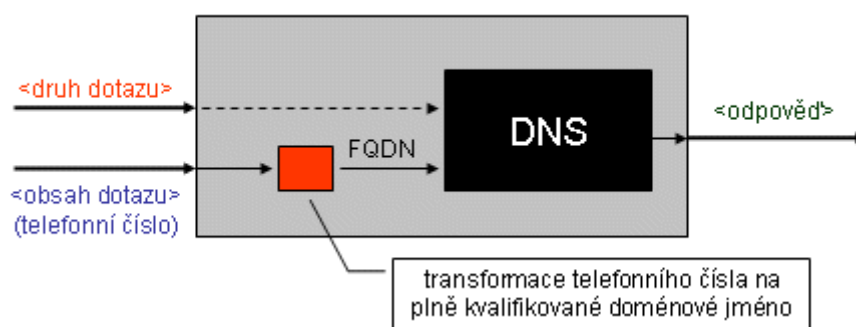
Obrázek 2.6 Vzorový příklad navázání SIP spojení pomocí ENUM (upraveno dle [7])

Na obrázku je vidět spojení mezi dvěma softphony Yate a Jitsi v mé praktické konfiguraci této práce. Na jedné straně je softphone Yate, který má telefonní klapku 777 s názvem stanice_1. Na druhé straně je softphone Jitsi, který má telefonní klapku 666 s názvem stanice_2. Yate chce uzavřít spojení s Jitsi, proto volá klapku 666. Požadavek je vyslán na SIP server, kde se telefonní číslo převede na plně kvantifikované doménové jméno 6.6.6.e164.arpa. Systém ENUM se poté ptá serveru DNS, jestli toto doménové jméno má ve své databázi. DNS odpovídá NAPTR záznamem, kde k doménovému jménu byl přidělen SIP URI záznam ve tvaru sip:666@158.196.244.205. SIP server má i odpovídající IP adresu volaného. SIP server nachází SIP server, kde SIP server už směruje volání k uživateli stanice_2.

2.6.1 Převod telefonního čísla na plně kvantifikované doménové jméno

Převod telefonního čísla v mezinárodním tvaru do kvantifikovaného doménového jména je následující. Jestliže máme v telefonním čísle nějaké nenumerníky znaky jako jsou například mezery, závorky či „+“, musíme je odstranit. Pokud má číslo tvar v devíti místném tvaru, musíme mu přidat prefix, který se určuje podle země ve kterém je používáno. Česká republika používá prefix 420. Prefix 420 se přidává do systému ENUM jestliže je veřejný. V případě této práce je využíváno telefonní číslo 666. Toto telefonní číslo se převede do plně kvantifikovaného tvaru tím způsobem, že se mezi každou číslici vloží tečka. Dostaneme tedy 6.6.6, poté jej zrcadlově obrátíme a přidáme zprava subdoménu .e164.arpa. Konečný tvar vypadá takto 6.6.6.e164.arpa.

Takto je telefonní číslo převedeno do plně kvantifikovaného tvaru a slouží jako vstup do systému DNS, které se nazývá FQDN



Obrázek 2.6.1 Detailní schéma funkce ENUMU [5]

3. Praktická konfigurace propojení ústředn Asterisk pomocí infrastrukturního ENUM stromu

3.1 Asterisk

Asterisk byl nainstalován na dva virtuální servery s IP adresama 158.196.244.204 a 158.196.244.205.

Asterisk je softwarová pobočková ústředna, která je vytvořena přímo na Vašem PC. Asterisk je open source aplikace, která znamená, že je to otevřený zdrojový kód, do kterého můžeme zapisovat. Asterisk pracuje na platformách Linuxu a Unixu, kde poskytuje veškeré vlastnosti PBX. Vytváří rozhraní telefonního hardwaru, softwaru a libovolné telefonní aplikace.

Instalace Asterisku je jednoduchá a instaluje se pod příkazem, ikdyž pro každou distribuci linuxu se tato instalace může lišit:

```
apt-get install asterisk
```

Po nainstalování Asterisku k němu můžeme připojit v příkazovém řádku parametrem `r` (running):

```
asterisk -rvvvvv
```

V této fázi byl Asterisk nainstalovaný a proto mohla začít konfigurace. Hlavními konfiguračními soubory v Asterisku jsou `sip.conf` a `extensions.conf`. Tyto dva soubory většinou najdeme ve složce `/etc/asterisk`.

3.1.1 sip.conf

Konfigurační soubor `sip.conf` je určený pro definici vlastností uživatelů na protokolu SIP. V této práci byly nadefinovány dvě stanice pod názvy `stanice_1` a `stanice_2`. Uváděný příklad je zkrácen o definování `stanice_2`, liší se pouze číselnými rozsahy. Jako signalizační protokol pro tuto práci byl použit protokol SIP. Příklad konfigurace `sip.conf`:

```
[general]
port=5060
context=vsvb
transport=udp
qualify=yes
svrlookup=yes
allowguest=yes
allow=ulaw
```

V sekci `[general]` se nachází všeobecné nastavení pro celý konfigurační soubor. UDP port byl nastaven na 5060, kde se port používá pro komunikaci SIP. Asterisk rozlišuje umístění zařízení do domén, tzn. kontextů. Pro jednoduchost byl využit pouze jediný,

3.Praktická konfigurace propojení ústředn Asterisk pomocí infrastrukturního ENUM stromu

pracovně nazvaný vsb. Při přenosu byl využit pouze protokol UDP, který byl nastaven v `transport=udp`. Nastavením `qualify=yes` nám Asterisk periodicky každých 60 sekund monitoruje stav peeringu mezi uživateli. Povolení pro volání doménových jmen na odchozí hovory bylo nastaveno `svrlookup=yes`. Pro povolení různorodých kodeků a pro správnou funkčnost Asterisku bylo nastaveno `allow` na vybrané kodeky, které budou využity při spojeních mezi telefony. Naopak pro zakázání kodeků byl použit `disallow`. Abychom nevyžadovali ověření na příchozí požadavky pro peer nastavíme `insecure=invite, port`. Celá sekce `[general]` se nachází v seznamu příloh.

Nyní byly nastaveny všechny obecné vlastnosti na PBX. Můžeme přejít k nastavování vlastností stanic. Základní definice účtu s názvem stanice_1 má následující tvar:

```
[stanice_1]
type=friend
secret=1234
userid=stanice_1 vsb <777>
host=dynamic
context=vsb
```

Type nám říká, jaký vtať bude mít stanice k Asterisku, jestliže budeme povolovat příchozí i odchozí hovory, jestliže ji nastavíme na `friend`. *Secret* bylo nastaveno heslo, které při registraci bude telefonní přístroj nebo softphone vyžadovat. Dále bylo nastaveno `userid`, jehož funkcí je zobrazení jména na displeji telefonu a jaké telefonní číslo(linka) se přiřadí. Podle toho na jakém telefonním přístroji bude pracovat, nastavíme i `host`. Kdybychom pracovali na stálém telefonním přístroji, nastavíme `static` tzn., že se IP adresa nebude měnit. Jelikož toto nastavení se může používat na různých místech nastavíme `dynamic`, čímž zajistíme, že přístroji se bude moct měnit IP adresa. Celé toto nastavení bylo zařazeno do kontextu `vsb`.

Stejně byla nastavena i druhá stanice_2, s tím rozdílem že její klapka byla 888. Celé nastavení stanice_1 i stanice_2 se nachází v sekci seznam příloh.

3.1.2 extensions.conf

Konfigurační soubor `extensions.conf`, který je základním souborem Asterisku, definuje chování volacího plánu (`dialplanu`) hovoru v ústředně. Podle něho se definuje jakým způsobem se budou hovory odehrávat. Soubor se nachází v `/etc/asterisk`

Základní syntaxí číslovacího plánu je:

```
exten => jméno, priorita, příkaz(parametry)
```

Kde `jméno` může obsahovat jméno nebo číslo, kam se chceme dovolat. Priorita můžeme nazvat tzv. Pořadové číslo příkazu. Začíná se prioritou 1 a dále pokračuje v číselné řadě. Jestliže budou mít prioritu 1, 2 a 5, vykonají se příkazy 1 a 2 a poté se hovor ukončí, protože nenašel prioritu 3 a 4. Prioritu můžeme taky zjednodušit pomocí písmene „n“. Když

3.Praktická konfigurace propojení ústředn Asterisk pomocí infrastrukturního ENUM stromu

Asterisk nalezne prioritu *n*, tak nahradí v paměti s předchozí prioritou číslo plus jedna. Příkazy určují a specifikují jednotlivé akce na jednotlivém kanále. Příkazy mají mnoho funkcí od přehrání zvuku, vyzvednutí hovoru nebo ukončení hovoru.

Začínalo se v sekci `[general]`, který obecně platí pro všechny další sekce:

```
[general]
static=yes
writeprotect=no
```

V sekci `[general]` bylo definováno `static=yes` a `writeprotect=no`, čímž bylo zamezeno přepisování dialplanu ze spuštěného Asterisku.

```
[globals]
```

Sekce `[globals]` využívá globální proměnné používané v Asterisku. Pro tuto práci je tato sekce nepotřebná.

```
[vsb]
exten => 777,1,Dial(SIP/stanice_1)
exten => 777,n,Hangup()
exten => 888,1,Dial(SIP/stanice_2)
exten => 888,n,Hangup()
```

V kontextu `vsb`, který byl definován v souboru `sip.conf` následně bylo definováno interní volání mezi dvěma stanicemi. První `exten` definoval číslo 777, který přepojoval hovor na uživatele `stanice_1` definován v `sip.conf`. Po ukončení hovoru se funkcí `Hangup()` hovor zavěsil. Stejně funguje volání na `stanice_2` s číslem 888.

```
exten => _XXX,1,Macro(enum,${EXTEN})
```

V tomto příkazu byl definován `pattern` odpovídající číslu. Podtržítka značí začátek odpovídajícího čísla a každé `X` nahrazuje číslo od 0 do 9. Tomuto `patternu` odpovídala telefonní čísla s třemi číslicemi. Používání makra `macro-enum` odpovídalo základním prováděným akcím, které byly provedeny při vytočení daného telefonního čísla. Do parametru `${EXTEN}` se předalo volané telefonní číslo.

```
[macro-enum]
exten => _s,1,NoOp(„Promenna : ${ARG1}“)
exten => _s,2,Dial(SIP/${ENUMLOOKUP (${ARG1},sip,1,e164.arpa)})
```

Byla nadefinována struktura `macro-enum`. Macra v Asterisku mají funkci zjednodušení a zpřehlednění psaní kódu. První řádek makra vypíše proměnnou, která je nyní zapsána v prvním argumentu `${ARG1}` a předala se pomocí parametru `${EXTEN}`. Úkolem

3.Praktická konfigurace propojení ústředn Asterisk pomocí infrastrukturního ENUM stromu

funkce `ENUMLOOKUP` je vyhledávání čísel (v tomto případě je číslo uloženo v proměnné `ARG1`) pomocí reverzního překladu v ENUM stromu a tato funkce vrací NAPTR záznam.

3.2 Bind

Bind je program pro implementaci DNS serveru v Internetu. Je součástí většiny linuxových distribucí. Program Bind byl nainstalován pomocí příkazu

```
apt-get install bind
```

Po nainstalování programu Bind je velmi důležité, aby byl nastaven SIP server, který se dotazuje do name serveru, který byl konfigurován. Změna byla provedena v souboru `/etc/resolv.conf`, kde byl nastaven nameserver 158.196.244.204. [2]

```
Nameserver 158.196.244.204
```

3.2.1 named.conf

V konfiguračním souboru `named.conf` byla vytvořena zóna `6.6.6.e164.arpa` pro reverzní překlad telefonního čísla 666. Telefonní číslo je v plném kvantifikovaném stavu. Zóna se odkazovala na soubor `db.666`.

```
zone "6.6.6.e164.arpa" IN {  
    type master;  
    file "/etc/bind/db.666";  
};
```

3.2.2 db.666

V souboru `db.666` byly nastaveny vlastnosti DNS serveru. Nejdůležitější částí však tohoto konfiguračního souboru je část NAPTR záznamu. NAPTR záznam přiřazuje pro dotaz na číslo 666 k němu odpovídající SIP URI adresu ve tvaru `sip:666@158.196.244.205`. V NAPTR záznamu byly zaslána IP adresa SIP serveru na do, kterého jsme se dotazovali. Na virtuálním serveru 158.196.244.204 bylo vytvořeno číslo 666, systém se dotázal do ENUM stromu a našel číslo 666 odpovídající NAPTR záznam. Poté vrátil SIP URI adresu ve tvaru `sip:666@158.196.244.205`. Podle této adresy virtuální server 158.196.244.205 vyhledal stanici s číslem 666. Celá konfigurace se nachází v sekci přílohy.

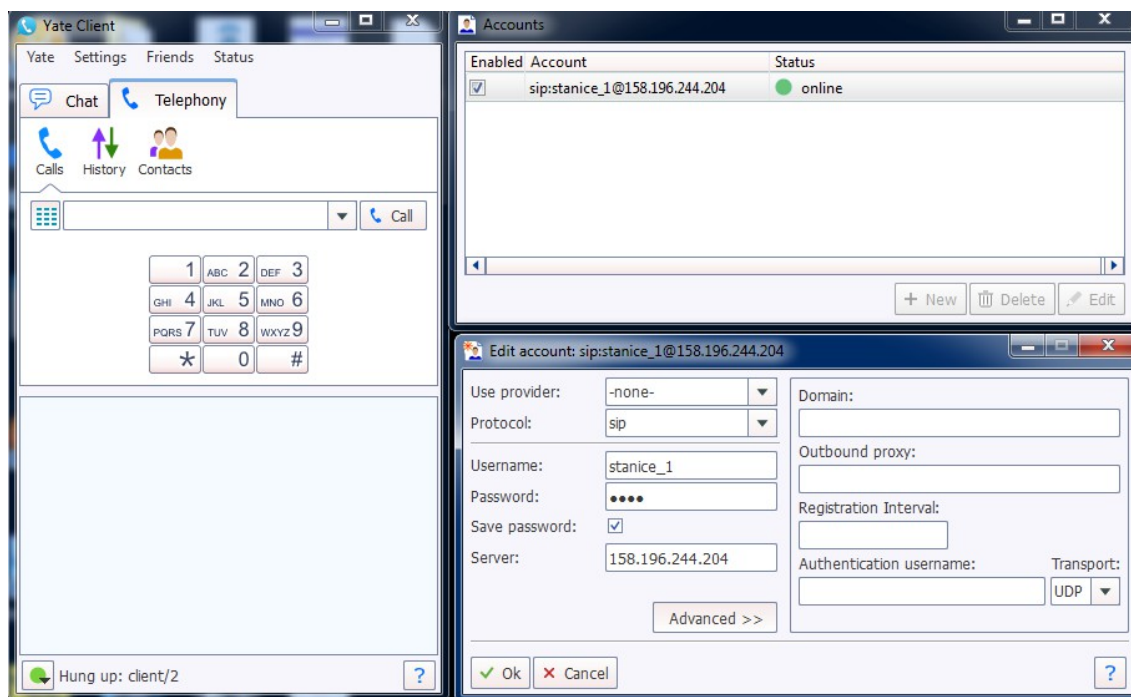
```
$ORIGIN 6.6.6.e164.arpa.
```

```
1 IN NAPTR 100 10 "u""sip+E2U" "!^.*$!sip:666@158.196.244.205!".
```

3.3 Registrace na softphonech

Byly nakonfigurovány oba dva virtuální servery. V dalším kroku byla otestována funkčnost spojení mezi servery. Pro tyto účely byly nainstalovány dva softphony, nebo-li softwarové telefony. Použity byly softphony Yate a Jitsi.

3.3.1 Yate



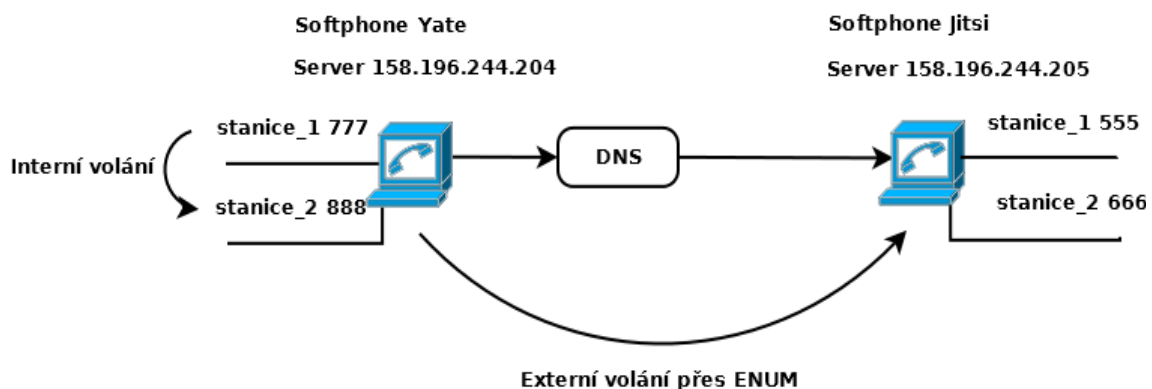
Obrázek 3.3.1 Yate, nastavení SIP účtu

Na obrázku můžeme vidět softphone Yate, kde byl účet registrován pod SIP URI adresou `sip:stanice_1@158.196.244.204`. Při registraci byly nastavovány stejné údaje jako v Asterisku konkrétně v konfiguračním souboru `sip.conf`. Obdobným způsobem byl nakonfigurován softwérový telefon Jitsi. Na druhém softphone Jitsi byl nastavován SIP URI adresa s `sip:stanice_2@158.196.244.205`.

3.4 Ověření funkčnosti - popis

V této kapitole byly nainstalovány na obou serverech programy Asterisk a Bind. Byla zajištěna komunikace interní. Na jednom serveru může být mezi dvěma stanicemi uzavřeno spojení, které bylo v této práci funkční. Také byla zajištěna komunikace externí a to přes systém ENUM tzn. Z jednoho serveru dokážeme uzavřít spojení na server druhý právě přes zmiňovaný systém ENUM. Interní komunikace byla nastavena na obou serverech. Na serveru 158.196.244.204 jsou stanice pojmenovány `stanice_1` a `stanice_2`, kde `stanice_1` má číselný rozsah 777 a `stanice_2` číselný rozsah 888. Na serveru 158.196.244.205 jsou stanice pojmenovány `stanice_1` a `stanice_2` s tím rozdílem, že `stanice_1` má číselný rozsah 555 a `stanice_2` číselný rozsah 666. Příklad externí komunikace je následovný na serveru 158.196.244.204 bylo voláno číslo 666, které přes systém ENUM se dovolalo na server 158.196.244.205, respektive na `stanice_2`. Tato externí komunikace byla otestována na

3. Praktická konfigurace propojení ústředěn Asterisk pomocí infrastrukturního ENUM stromu softphonech Yate a Jitsi, a byla funkční. Ověření funkčnosti nalezeneme v kapitole pět. Na obrázku můžeme vidět schéma zapojení v této práci.



Obrázek 3.4 Schéma zapojení

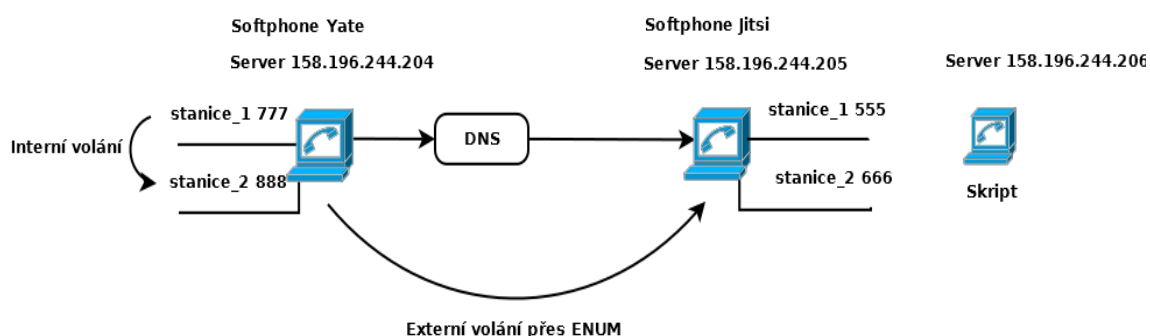
4. Automatizace řešení pomocí instalačního skriptu

V předchozí části této práce byly nakonfigurovány dvě stanice, které byly propojeny pomocí systému ENUM. Tento krok byl zautomatizován vytvořením skriptu. Skript byl vytvořen tak, aby dal uživateli možnost nazvat si stanice a jejich číselné rozsahy podle své potřeby, a to bylo účelem tohoto skriptu. Skript byl navržen takovým způsobem, aby uživatel povinně zadal všechny čtyři parametry. Jestliže takto uživatel neudělal, skript ho vybědl chybovou hláškou „Chybí parametr“. Pokud uživatel zadal špatně jeden z parametrů, byla mu vypsaná nápověda, podle které se měl uživatel řídit. Skript je navržen tak, aby ošetřoval chybné zadání vstupních proměnných.

Vstupní parametry se ukládaly do dynamických proměnných, které se po každém spuštění skriptu přepisovaly. Skript byl navržen tak, aby v konfiguračních souborech vyhledal klíčové slovo a přepsal jej právě dynamickým parametrem. Očekávanost klíčového slova je na místech, kde se budou přepisovat názvy stanic nebo číselné rozsahy.

Skript vytvářel zálohu pro konfigurační soubory, které byly přepisovány. Originální konfigurační soubor byl vytvořen na svém místě, která určovala instalace. Skript tento konfigurační soubor zkopíroval na místo, které bylo ve skriptu zadáno. Poté jej skript přepsal zadanými parametry od uživatele. Skript byl ošetřen tak, aby jeho originální konfigurace byla zálohována a přepsané konfigurační soubory byly v pracovním režimu.

Skript byl vytvořen pro novou registraci uživatelů na novém serveru a to je hlavní důvod vytvoření tohoto skriptu. Pokud uživatel spustí tento skript získá pohodlnou registraci na svých pobočkových ústřednách Asterisku. Ve skriptu byl nadefinován systém ENUM, tudíž uživatel může volat i externě. Na novém serveru předpokládáme instalaci Asterisku a Bindu. Skript byl otestován na serveru 158.196.244.205, a je funkční a celý se nachází v přílohách. Na obrázku můžeme vidět nový server 158.196.244.206, na kterém můžeme využít skript pro konfiguraci volání.



Obrázek 4 Využití skriptu na novém serveru

5. Ověření funkčnosti propojení pomocí rozboru signalizace

5.1 Wireshark

Wireshark je program umožňující analyzovat, zachytávat a překládat všechny protokoly, které počítač přijímá či odesílá. Wireshark nám nabízí mnoho funkcí, které uživatel ocení např. u sledování grafického výstupu z provozu sítě nebo filtrace paketů protokolu.

5.2 Rozbor ENUM

V rozboru pomocí systému ENUM byly využity již předdefinované softphony Yate a Jitsi. Softphone Yate a Jitsi byly propojeny externě přes ENUM. V příkazovém řádku na serverech 158.196.244.204 a 158.196.244.205 byl zadán příkaz `tcpdump`, který odchytil všechny protokoly, které spojení využívalo. Struktura příkazu vypadá následovně :

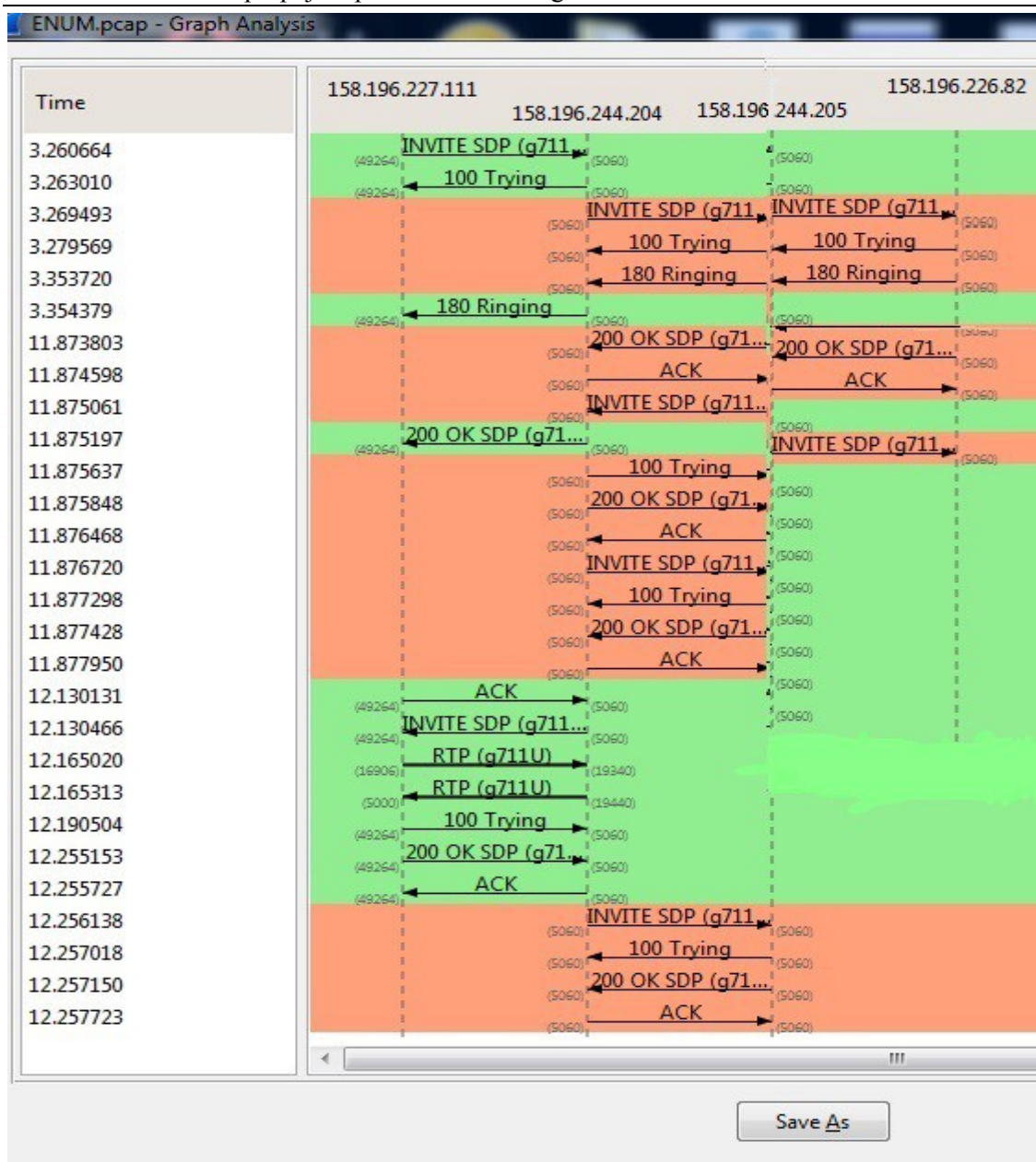
```
tcpdump -i any -vv -w /tmp/pcap.pcap
```

Příkaz `tcpdump` poslouchal na všech portech. Parametr `-vv` zaručoval podrobnost výstupu. Parametr `-w` zapisoval síťovou komunikaci do souboru `/tmp/pcap.pcap`, kde přípona `pcap` je spustitelný soubor v programu Wireshark.

Odchyťování komunikace bylo nastaveno a hovor může proběhnout. Do softphonu Yate bylo zadáno telefonní číslo softphonu Jitsi, což v našem případě je telefonní číslo 666 a hovor byl spuštěn. Když softphone Jitsi začal zvonit, hovor byl námi přijat a na pár sekund bylo spojení aktivní, poté byl hovor zavěšen. Příkaz `tcpdump` v příkazovém řádku byl ukončen a soubor byl stahován pomocí některého z programů, který umožňuje komunikaci mezi vzdáleným serverem a našim počítačem. Pro tuto operaci byl použit program WinSCP. Takto bylo postupováno, jak na serveru 158.196.244.204 tak i na server 158.196.244.205

Na obrázku můžeme vidět vyfiltrovanou komunikaci protokolem SIP mezi stanicemi Yate a Jitsi. Softphonu Yate byla přidělena IP adresa 158.196.227.111 a softphonu Jitsi byla přidělena IP adresa 158.196.226.82.

5. Ověření funkčnosti propojení pomocí rozboru signalizace



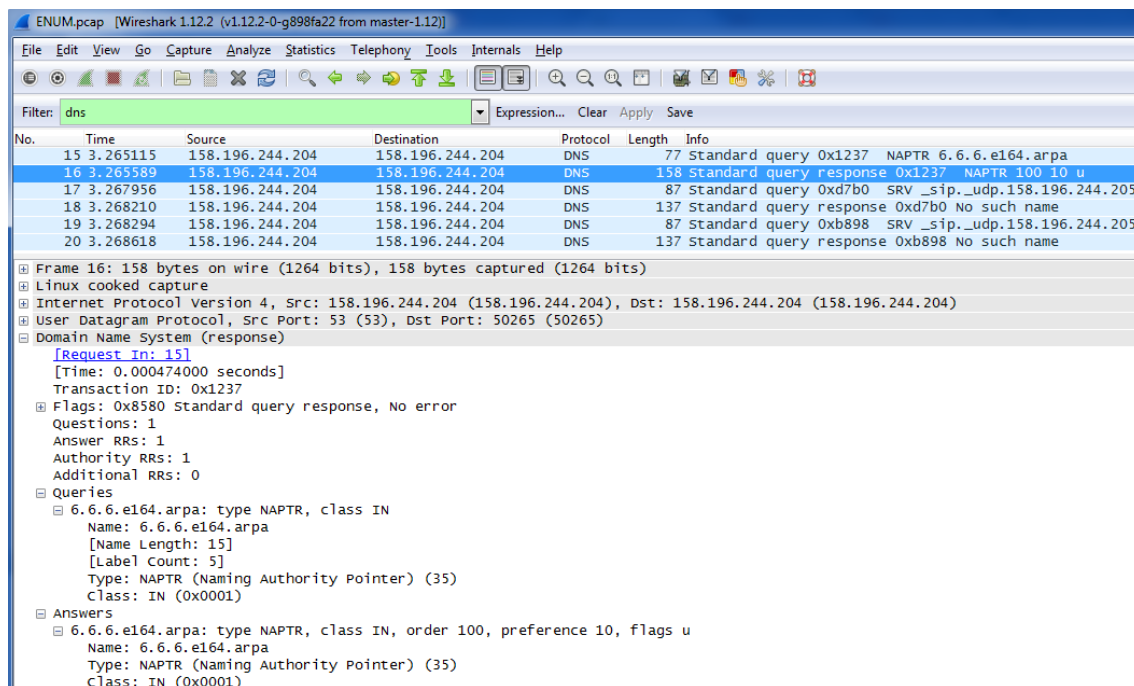
Obrázek 5.2.1 Rozbor signalizace v programu Wireshark

Softphone Yate volal telefonní číslo 666 softphonu Jitsi. Yate žádal o inicializaci spojení zprávou INVITE na SIP server 158.196.244.204. Server 204 mu odpovídal zprávou 100 Trying, která zahajovala zpracování žádosti INVITE. Server 204 stejně žádal server 205 o inicializaci spojení. Server 205 poté žádal o inicializaci spojení softphon Jitsi. Dále následovala zpráva 180 Ringing, která označovala vyzvánění volaného účastníka. V okamžiku, kdy volaný softphon Jitsi přijal hovor, vyšle se zpráva 200 OK, která potvrzuje zpracování zprávy INVITE. Poté softphone vyšle potvrzovací zprávu ACK. Nyní byly vyslány zprávy Re-Invite, které upřesňují telefonní spojení tím, že se domlouvaly na použití kodeku při hovoru atd. Tyto zprávy byly potvrzovány softphony zprávy 200 OK a následně přijetí odpovědi zprávou ACK. Poté co, se ústředny domluvily na podrobnostem komunikace mohly začít vysílat své data. Přenášena data

5. Ověření funkčnosti propojení pomocí rozboru signalizace

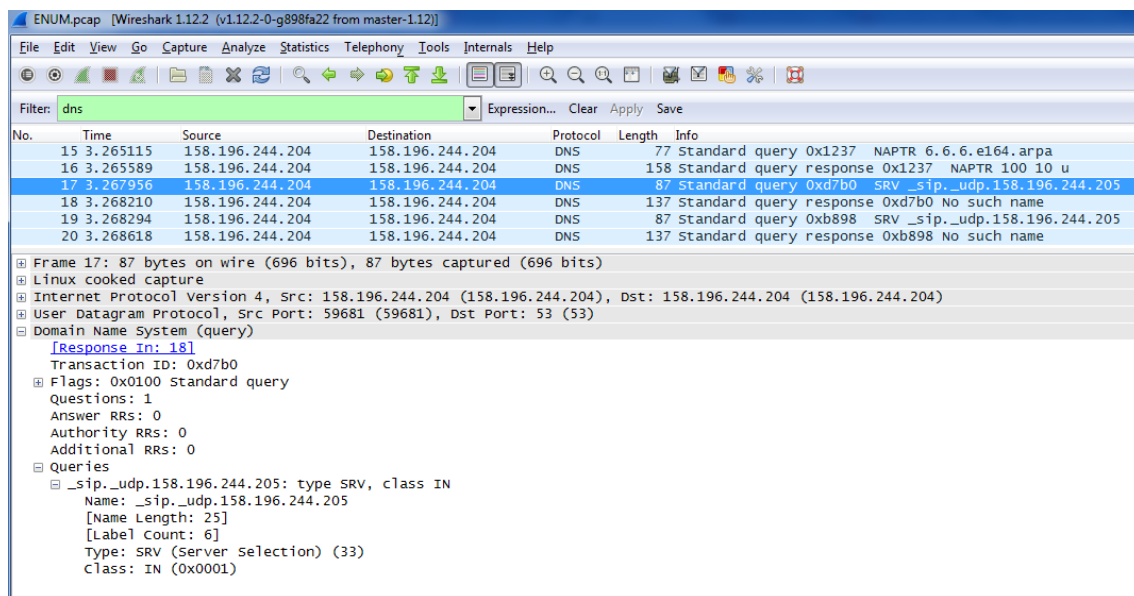
se vysílala přes protokol RTP, který slouží pro přenášení dat v reálném čase. Pro ukončení hovoru softphoned Jitsi vyslal zprávu BYE, která se potvrdila zprávou 200 OK.

Vyfiltrováno bylo spojení i přes DNS server, kde se dotazovalo na plně kvantifikované doménové jméno 6.6.6.e164.arpa. DNS server odpověděl zprávou _sip._udp.158.196.244.205, podle této zprávy Asterisk už věděl kam má směřovat volání. Na obrázku můžeme vidět dotaz do DNS serveru.



Obrázek 5.2.2 Dotaz do DNS serveru

Na obrázku můžete vidět odpověď DNS serveru



Obrázek 5.2.3 Odpověď DNS serveru

Závěr

Tato práce se zabývala především systémem ENUM. Používáním tohoto systému mají uživatelé možnost získat zajímavé výhody při vzájemné komunikaci. Pomocí služby ENUM mohou uživatelé bezplatně vést telefonní hovory přes Internet namísto standartně placené telefonní sítě, z toho vyplývá hlavní výhoda systému, a to je úspora nákladů za telefonní hovory. Telefonní operátoři mají k systému ENUM až aktivní odpor, protože vnímají ENUM jako hrozbu. ENUM směřuje k tomu, aby se uživatelé vyhýbali operátorským sítím. Podle mého názoru bude mít tudíž ENUM do budoucna velmi pomalý nástup.

Cílem práce bylo navrhnout, zautomatizovat a ověřit funkční spojení mezi pobočkovými ústřednami Asterisků přes systém ENUM.

V prvních dvou kapitolách byly popsány teoretické poznatky technologie VoIP a systém ENUM. V třetí kapitole bylo navrženo spojení mezi pobočkovými ústřednami Asterisků. Interní spojení fungovalo na jednom serveru, kde dva uživatelé se dovolali mezi sebou. Externí spojení fungovalo přes systém ENUM, kdy se jeden uživatel na jednom serveru dovolal uživateli na dalším serveru. Oba tyto způsoby byly testovány a byly funkční. Ve čtvrté kapitole byl navržen skript. Hlavní důvod vytvoření skriptu byla nová registrace uživatelů a jejich telefonních čísel uživatelem na novém serveru. Tento skript byl spustitelný na serveru a byl funkční. V páté kapitole byla rozebrána a vysvětlena komunikace spojení přes systém ENUM z pohledu signalizace v programu Wireshark.

Hlavním přínosem této práce bylo seznámení se s systémem ENUM a jeho praktické implementace do infrastrukturního prostředí.

Použitá literatura

- [1] PETERKA, Jiří. Z historie internetové telefonie (nejen) v ČR. *Earchiv.cz* [online]. 2011 [cit. 2015-04-03]. Dostupné z: <http://www.earchiv.cz/b06/b0401006.php3>
- [2] Doc. Ing. VOZŇÁK,, Miroslav Ph.D. *Voice over IP*. Ostrava, 2012. ISBN 978-80-248-1828-3. VŠB-TU Ostrava.
- [3] KRESTA, Jaroslav. *ENUM - E.164 Number Mapping*. Brno, 2007. Bakalářská práce. Masarykova univerzita v Brně.
- [4] PETERKA, Jiří. ENUM by si zasloužil vlastní číselnou řadu. *Earchiv.cz* [online]. 2011 [cit. 2015-04-03]. Dostupné z: <http://www.earchiv.cz/b06/b1003001.php3>
- [5] PETERKA, Jiří. Co je ENUM ?. *Lupa.cz* [online]. 2006 [cit. 2015-04-08]. Dostupné z: <http://www.lupa.cz/clanky/co-je-enum/>
- [6] ENUM SLAVÍ PRVNÍ ROK V KOMERČNÍM PROVOZU. *Nic.cz* [online]. 2015 [cit. 2015-04-03]. Dostupné z: <https://www.nic.cz/page/439/enum-slavi-prvni-rok-v-komercnim-provozu/>
- [7] Softwarová ústředna Asterisk a ENUM. *Cs.vsb.cz* [online]. [cit. 14.4.2014]. Dostupné z: <http://www.cs.vsb.cz/grygarek/TPS/projekty/0607Z/ENUM/enum.html>
- [8] SIP: Session Initiation Protocol. *Ietf.org* [online]. 2002 [cit. 2015-04-30]. Dostupné z: <https://www.ietf.org/rfc/rfc3261.txt>

Seznam příloh

Asterisk

sip.conf

```
[general]
context=vsb
port=5060
transport=udp
qualify=yes
svrlookup=yes
allowguest=yes
allow=ulaw
allow=alaw
allow=gsm
allow=h263
allow=h263p
allow=h264
insecure=port,invite

[stanice_1]
type=friend
secret=1234
dtmfmode=rfc2833
userid=stanice_1 vsb <777>
host=dynamic
deny=0.0.0.0/0.0.0.0
permit=158.196.0.0/255.255.0.0
context=vsb

[stanice_1]
type=friend
secret=1234
```

```
dtmfmode=rfc2833
userid=stanice_1 vsb <888>
host=dynamic
deny=0.0.0.0/0.0.0.0
permit=158.196.0.0/255.255.0.0
context=vsb
```

extensions.conf

```
[general]
static=yes
writeprotect=no
clearglobalvars=no

[globals]
[vsb]
exten => 777,1,Dial(SIP/stanice_1)
exten => 777,n,Hangup()
exten => 888,1,Dial(SIP/stanice_2)
exten => 888,n,Hangup()

exten => _XXX,1,Macro(enum,${EXTEN})
[macro-enum]
exten => _s,1,NoOp(„Promenna : ${ARG1}“)
exten => _s,2,Dial(SIP/${ENUMLOOKUP (${ARG1},sip,1,e164.arpa)})
```

Bind

named.conf

```
zone "6.6.6.e164.arpa" IN {
    type master;
    file "/etc/bind/db.666";
};
```

db.666

```
$TTL      86400
@         IN      SOA      localhost. root.localhost.      (
                                9              ; serial
                                28800          ; Refresh
                                14400          ; Retry
                                3600000        ; Expire
                                86400 )       ; Minimum
        IN      NS      localhost.

$ORIGIN    6.6.6.e164.arpa.
1 IN NAPTR 100 10 "u" "sip+E2U" "!^.*$!sip:666@158.196.244.205!"
.
```

Script

```
#!/bin/sh

IP=$(ifconfig eth0 | awk '{print $2}' | grep -E -o "([0-9]{1,3})[\.]{3}[0-9]{1,3}")
echo $IP

usage() { echo "Usage: $0 [-i<Stanice_1:>] [-p<Klapka_1:>] [-n<Stanice_2:>] [-m<Klapka_2:>]"1>&2 ; exit 1; }

while getopts ":i:p:n:m:" o;do
    case $o in
        i) i=${OPTARG}
            ;;
        p) p=${OPTARG}
            ;;
        n) n=${OPTARG}
            ;;
        m) m=${OPTARG}
```

```

;;
*)
echo "Chybi paramaetr: -$OPTARG" >&2
;;
esac
done
shift $((OPTIND-1))

if [ -z "${i}" ] || [ -z "${n}" ] || [ -z "${p}" ] || [ -z "${m}" ]; then
usage
fi

###
#prikaz echo "123" | sed -r 's#([0-9]{1})([0-9]{1})([0-9]{1})#\3\2\1#' | sed 's/\(.\\{1\\}\)/\1./g'
###
p_opak=$ echo "$p" | sed -r 's#([0-9]{1})([0-9]{1})([0-9]{1})#\3\2\1#' | sed 's/\(.\\{1\\}\)/\1./g'
echo $p_opak

m_opak=$ echo "$m" | sed -r 's#([0-9]{1})([0-9]{1})([0-9]{1})#\3\2\1#' | sed 's/\(.\\{1\\}\)/\1./g'
echo $m_opak

if [ -f /etc/asterisk/sip.conf ]
then
mkdir -p $BASE/backup/etc
mkdir -p $BASE/backup/etc/asterisk/

cp /etc/asterisk/sip.conf
$BASE/backup/etc/asterisk/sip.conf_backup_from_installation_script
unlink /etc/asterisk/sip.conf

fi

mkdir -p $BASE/etc

```

```
cp $BASE/install/sip.conf $BASE/etc/sip.conf
```

```
echo "First part"
```

```
sed -i "s/ST1/$i/g" $BASE/etc/sip.conf
```

```
sed -i "s/KL1/$p/g" $BASE/etc/sip.conf
```

```
sed -i "s/ST2/$n/g" $BASE/etc/sip.conf
```

```
sed -i "s/KL2/$m/g" $BASE/etc/sip.conf
```

```
if [ -f /etc/asterisk/extensions.conf ]
```

```
then
```

```
mkdir -p $BASE/backup/etc
```

```
mkdir -p $BASE/backup/etc/asterisk/
```

```
cp /etc/asterisk/extensions.conf  
$BASE/backup/etc/asterisk/extensions.conf_backup_from_installati  
on_script
```

```
unlink /etc/asterisk/extensions.conf
```

```
fi
```

```
mkdir -p $BASE/etc
```

```
cp $BASE/install/extensions.conf $BASE/etc/extensions.conf
```

```
echo "Second part"
```

```
sed -i "s/ST1/$i/g" $BASE/etc/extensions.conf
```

```
sed -i "s/KL1/$p/g" $BASE/etc/extensions.conf
```

```
sed -i "s/ST2/$n/g" $BASE/etc/extensions.conf
```

```
sed -i "s/KL2/$m/g" $BASE/etc/extensions.conf
```

```
if [ -f /conf/named.conf ]
```

```
then
```

```
mkdir -p $BASE/backup/etc
```

```
mkdir -p $BASE/backup/etc/bind
```

```
cp /conf/named.conf
$BASE/backup/etc/bind/named.conf_backup_from_installation_script
unlink /conf/named.conf
```

```
fi
mkdir -p $BASE/etc
cp $BASE/install/named.conf $BASE/etc/named.conf
```

```
echo "Third part"
sed -i "s/KL1/$p_opak/g" $BASE/etc/named.conf
sed -i "s/KL2/$m_opak/g" $BASE/etc/named.conf
```

```
if [ -f /conf/db.888 ]
then
mkdir -p $BASE/backup/etc
mkdir -p $BASE/backup/etc/bind
cp /conf/db.888
$BASE/backup/etc/bind/db.888_backup_from_installation_script
unlink /conf/db.888
```

```
fi
mkdir -p $BASE/etc
cp $BASE/install/db.888 $BASE/etc/db.888
```

```
echo "Four part"
sed -i "s/KL_1/$p_opak/g" $BASE/etc/db.888
sed -i "s/KL1/$p/g" $BASE/etc/db.888
sed -i "s/IP/$IP/g" $BASE/etc/db.888
```

```
if [ -f /conf/db.999 ]
then
mkdir -p $BASE/backup/etc
```

```
mkdir -p $BASE/backup/etc/bind
cp /conf/db.999
$BASE/backup/etc/bind/db.999_backup_from_installation_script
unlink /conf/db.999

fi

mkdir -p $BASE/etc
cp $BASE/install/db.999 $BASE/etc/db.999

echo "Five part"
sed -i "s/KL_2/$m_opak/g" $BASE/etc/db.999
sed -i "s/KL2/$m/g" $BASE/etc/db.999
sed -i "s/IP/$IP/g" $BASE/etc/db.999

cp $BASE/etc/sip.conf /etc/asterisk/sip.conf
cp $BASE/etc/extensions.conf /etc/asterisk/extensions.conf

cp $BASE/etc/named.conf conf/named.conf
cp $BASE/etc/db.888 conf/db.888
cp $BASE/etc/db.999 conf/db.999

/etc/init.d/asterisk restart
```